

带有透明阴影效果的文本框

— TikZ 学习笔记 (2)

lyanry@gmail.com

September 4, 2007

这是我学习 TikZ 的一篇笔记, 逐步讲解如何使用 TikZ 绘制带有透明阴影效果的文本框, 将该技巧应用于 Beamer 演示文档, 可以让你的讲稿增色许多。

原始资料来源于: <http://www.fauskes.net/pgftikzexamples/transparent-shadows/>

1 做一些准备

首先建立一个完整的 L^AT_EX 源文档环境:

```
\documentclass{article}
\usepackage[absolute,overlay]{textpos}
\usepackage{tikz}

\begin{document}

\end{document}
```

`textpos` 宏包是用来实现文本盒的绝对坐标定位的, 设定 `absolute` 与 `overlay` 选项的目的是让文本块可以在文档页面上使用绝对坐标定位, 并可保证文本盒不会被其他元素所覆盖。

然后在文档的导言区内定义一些用于表示某种长度的 L^AT_EX 命令:

```
\newlength{\boxw}
\newlength{\boxh}
\newlength{\boxroundness}
\newlength{\shadowsize}
\newlength{\tmpa}
```

`\boxw` 与 `\boxh` 分别表示文本盒的宽度与高度, `\boxroundness` 表示文本盒圆角的半径大小。`\shadowsize` 表示文本盒阴影的宽度。`\tmpa` 在带有阴影的文本盒的绘制过程中起到一个中间变量的作用。

然后设定 `\shadowsize` 与 `\boxroundness` 的长度:

```
\setlength{\shadowsize}{6pt}
\setlength{\boxroundness}{3pt}
```

2 创建一个文本盒

关于 $\text{T}_{\text{E}}\text{X}/\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ 文本盒的基础知识, 这里不费笔墨¹。

考虑到代码复用, 我们还是采用自定义命令的方式, 将文本盒环境的构建封装为一个专用的 $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ 命令。

首先, 要声明一个盒子:

```
\newsavebox{\shadowblockbox}
```

然后我们在这个盒子里放置一个 `minipage`, 尝试将其宽度定义为文本行宽度的一半, 盒子中填充文本 “Test shadowblockbox”:

```
\begin{lrbox}{\shadowblockbox}
\begin{minipage}{.5\textwidth}
Test shadowblockbox.
\end{minipage}
\end{lrbox}
```

然后就可以在正文中使用这个盒子:

```
\usebox{\shadowblockbox}
```

以上是 $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ 盒子复用过程, 即: 声明、定义、使用。就是说一旦你定义了一个盒子, 可以在任何需要显示盒子的地方 `\usebox{\boxname}`。

但是我们不仅仅是要复用盒子, 而是要定义一个命令, 让它接受 `minipage` 宽度值为参数, 而不是硬性的设定其参数为 `.5\textwidth` 之类的数值。这一次, 玩点更复杂的, 不是定义一个命令, 而是定义一个环境:

```
\newenvironment{shadowblock}[1]{
    \begin{lrbox}{\shadowblockbox}
    \begin{minipage}{\#1}
}
{
    \end{minipage}
    \end{lrbox}
}
```

有了 `shadowblock` 环境, 就可以将 `\shadowblockbox` 中的 `minipage` 宽度作为环境必须的一个参数传入, 从而定义了一个内部为 `minipage` 的盒子环境:

```
\newenvironment{shadowblock}[1]
{\begin{lrbox}{\shadowblockbox}\begin{minipage}{\#1}}
{\end{minipage}\end{lrbox}}
```

下面给出一个关于 `shadowblock` 环境的用例:

```
\begin{shadowblock}{5cm}
Test shadowblock environment.
\end{shadowblock}
```

¹关于 $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ 文本盒的知识可以参考《 $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ 入门与提高》第二版 109-114 页

3 绘制文本盒的阴影

前面的内容一直是定义 \LaTeX 变量、环境之类的，与 \TikZ 没有什么关系，但是也不要轻视这些工作，起码它可以让你认识或回顾一下 \LaTeX 的一些较为高级的内容。

下面我们会对 `shadowblock` 环境的 `end` 部分进行修改，使用 \TikZ 绘制出盒子的阴影。

首先要对第一小节中所定义的长度命令进行赋值，目的是基于已定义的 `\shadowblockbox` 盒子参数去初始化带有透明阴影的文本盒参数：

```
\newenvironment{shadowblock}[1]
{\begin{lrbox}{\shadowblockbox}\begin{minipage}{#1}}
{\end{minipage}\end{lrbox}}
\settowidth{\boxw}{\usebox{\shadowblockbox}}
\settodepth{\tmpa}{\usebox{\shadowblockbox}}
\settoheight{\boxh}{\usebox{\shadowblockbox}}
\addtolength{\boxh}{\tmpa}
\addtolength{\boxw}{\boxroundness * 2}
\addtolength{\boxh}{\boxroundness * 2}
}
```

通过 `\setto...` 与 `\addto...` 命令，就可以将 `\shadowblockbox` 盒子的宽度、高度值分别赋给 `\boxw`、`\boxh`。`\shadowblockbox` 的深度赋给“中间变量”`\tmpa`，然后再将该值加到 `\boxh` 上。由于我们是想让文本盒带有圆角，为了保证盒子内部内容不会被圆角区裁剪，需要对盒子的长、宽再增加 `\boxroundness * 2` 大小。

下面在 `shadowblock` 环境的 `end` 部分建立一个 `tikzpicture` 环境，并填充一个半透明的圆角矩形：

```
\newenvironment{shadowblock}[1]
{\begin{lrbox}{\shadowblockbox}\begin{minipage}{#1}}
{\end{minipage}\end{lrbox}}
\settowidth{\boxw}{\usebox{\shadowblockbox}}
\settodepth{\tmpa}{\usebox{\shadowblockbox}}
\settoheight{\boxh}{\usebox{\shadowblockbox}}
\addtolength{\boxh}{\tmpa}
\addtolength{\boxw}{\boxroundness * 2}
\addtolength{\boxh}{\boxroundness * 2}

\begin{tikzpicture}
  \foreach \x in {0,.05,...,1}
  {
    \setlength{\tmpa}{\shadowsize * \real{\x}}
    \fill[xshift=\shadowsize - 1pt,yshift=-\shadowsize + 1pt,
      black,opacity=.04,rounded corners=\boxroundness]
      (\tmpa, \tmpa) rectangle +(\boxw - 2\tmpa,
        \boxh - 2\tmpa);
  }
}
```

```
\end{tikzpicture}}
}
```



图 1 阴影矩形

我们在 `tikzpicture` 环境中使用了一个循环语句来实现多个半透明矩形的绘制，这些半透明矩形是由外向内收敛的，它们叠加到一起就可以实现图 1 所示的外围透明度较高而内部透明度较低的矩形效果。如果不甚明白这些代码的作用，可以采用下面代码将循环数的步进值增大一些，效果见图 2 所示。

```
\begin{tikzpicture}
  \foreach \x in {0,.5,...,1}
  {
    \setlength{\tmpa}{\shadowsize * \real{\x}}
    \fill[xshift=\shadowsize - 1pt,yshift=-\shadowsize + 1pt,
          black,opacity=.04,rounded corners=\boxroundness]
          (\tmpa, \tmpa) rectangle +(\boxw - 2\tmpa,
          \boxh - 2\tmpa);
  }
\end{tikzpicture}}
```

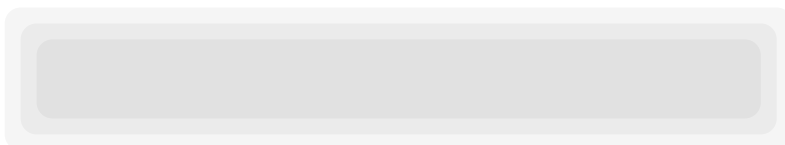


图 2 阴影矩形叠加效果

绘制阴影矩形的代码中，用到了一个 `\real` 命令，我没有查到这个命令是 \TeX 命令还是 \LaTeX 命令，它的作用是将一个用来表示数值的文本变为实数值，然后才能参与数学运算。

在填充阴影矩形时，设定坐标系沿 x 轴正向偏移了 `\shadowsize - 1pt`，沿 y 负向偏移了 `\shadowsize - 1pt`。在下一步绘制文本盒时，阴影矩形相对文本盒矩形就是向右下方偏移的，这样就可以产生文本盒的阴影效果。

4 绘制文本盒

下面绘制一个淡黄色背景的文本盒：

```
\newenvironment{shadowblock}[1]
{\begin{lrbox}{\shadowblockbox}\begin{minipage}{\#1}}
```

```

\end{minipage}\end{lrbox}
\settowidth{\boxw}{\usebox{\shadowblockbox}}
\settodepth{\tmpa}{\usebox{\shadowblockbox}}
\settoheight{\boxh}{\usebox{\shadowblockbox}}
\addtolength{\boxh}{\tmpa}
\addtolength{\boxw}{\boxroundness * 2}
\addtolength{\boxh}{\boxroundness * 2}

\begin{tikzpicture}
  \foreach \x in {0,.05,...,1}
  {
    \setlength{\tmpa}{\shadowsize * \real{\x}}
    \fill[xshift=\shadowsize - 1pt,yshift=-\shadowsize + 1pt,
          black,opacity=.04,rounded corners=\boxroundness]
      (\tmpa, \tmpa) rectangle +(\boxw - 2\tmpa,
                                \boxh - 2\tmpa);
    \filldraw[fill=yellow!50, draw=black!50, rounded corners=\boxroundness]
      (0, 0) rectangle (\boxw, \boxh);
  }
\end{tikzpicture}
}

```

文本盒与阴影矩形叠加后,就可以呈现如图 3 所示的文本盒的阴影效果。



图 3 文本盒阴影效果

5 文本显示

以上全部工作只是绘制了一个带有阴影的文本盒子,这一步要将盒子中的文本内容放置在 TikZ 结点中显示出来。

```

\newenvironment{shadowblock}[1]
{
  \begin{lrbox}{\shadowblockbox}\begin{minipage}{#1}
  \end{minipage}\end{lrbox}
  \settowidth{\boxw}{\usebox{\shadowblockbox}}
  \settodepth{\tmpa}{\usebox{\shadowblockbox}}
  \settoheight{\boxh}{\usebox{\shadowblockbox}}
  \addtolength{\boxh}{\tmpa}
  \addtolength{\boxw}{\boxroundness * 2}

```

```

\addtolength{\boxh}{\boxroundness * 2}

\begin{tikzpicture}
  \foreach \x in {0,.05,...,1}
  {
    \setlength{\tmpa}{\shadowsize * \real{\x}}
    \fill[xshift=\shadowsize - 1pt,yshift=-\shadowsize + 1pt,
          black,opacity=.04,rounded corners=\boxroundness]
      (\tmpa, \tmpa) rectangle +(\boxw - 2\tmpa,
                                \boxh - 2\tmpa);
    \filldraw[fill=yellow!50, draw=black!50, rounded corners=\boxroundness]
      (0, 0) rectangle (\boxw, \boxh);
    \draw node[xshift=\boxroundness,yshift=\boxroundness,
               inner sep=0pt,outer sep=0pt,anchor=south west]
      (0,0) {\usebox{\shadowblockbox}};
  }
\end{tikzpicture}
}

```

以上代码，需要注意的一点是绘制结点时指定了其锚点为 `south west`，也就是说让结点矩形的左下侧顶点位于坐标原点，这样可以保证 `\shadowblockbox` 盒子与前面绘制的文本盒相重合。

6 透明阴影效果的文本盒的使用

上面的所有工作都是围绕一个 `shadowblock` 环境定义开展的，现在这个环境已经定义完毕。要 `shadowblock` 环境所定义的文本盒，首先要新建一个盒子，在其中调入 `shadowblock` 环境：

```

\newsavebox{\mybox}

\begin{lrbox}{\mybox}
\begin{shadowblock}{4cm}
This is a block of text with a transparent shadow !
\end{shadowblock}
\end{lrbox}

```

然后可以在文档的不同位置使用 `\usebox{\mybox}` 命令重复使用盒子。

也可以利用 `textpos` 宏包提供的文本块定位功能，在文档的任意位置显示盒子：

```

\begin{textblock}{4cm}{2, 5.2}
\begin{shadowblock}{4cm}
And a small one on top of it.
\end{shadowblock}
\end{textblock}

```

图 4 显示了本节用例效果。

This is a block of text with a transparent shadow!

And a small one on top of it.

图 4 文本盒阴影效果

结语

TikZ 可以与 **L^AT_EX** 取得很好的结合。在 **L^AT_EX** 中如果能够灵活使用 **TikZ**, 可以取得许多梦幻般的效果。所以, 我们在学习 **L^AT_EX** 的过程中, 一定要积极地让一些宏包所提供的特殊效果融合到 **L^AT_EX** 文档中, 也省的人家指责 **L^AT_EX** 排版千篇一律。